



NAVAL POSTGRADUATE SCHOOL

MONTEREY, CALIFORNIA

THESIS

**SEGMENTED MIRROR TELESCOPE MODEL AND
SIMULATION**

by

Travis W. Axtell

June 2011

Thesis Co-Advisors:

Roberto Cristi
Jae-Jun Kim

Approved for public release; distribution is unlimited

THIS PAGE INTENTIONALLY LEFT BLANK

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

The public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden to Department of Defense, Washington Headquarters Services, Directorate for Information Operations and Reports (0704-0188), 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to any penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number. **PLEASE DO NOT RETURN YOUR FORM TO THE ABOVE ADDRESS.**

1. REPORT DATE (DD-MM-YYYY) 13-6-2011			2. REPORT TYPE Master's Thesis		3. DATES COVERED (From — To) 2010-09-01—2011-06-17	
4. TITLE AND SUBTITLE Segmented Mirror Telescope Model and Simulation					5a. CONTRACT NUMBER	
					5b. GRANT NUMBER	
					5c. PROGRAM ELEMENT NUMBER	
6. AUTHOR(S) Travis W. Axtell					5d. PROJECT NUMBER	
					5e. TASK NUMBER	
					5f. WORK UNIT NUMBER	
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Naval Postgraduate School Monterey, CA 93943					8. PERFORMING ORGANIZATION REPORT NUMBER	
9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES) Department of the Navy					10. SPONSOR/MONITOR'S ACRONYM(S)	
					11. SPONSOR/MONITOR'S REPORT NUMBER(S)	
12. DISTRIBUTION / AVAILABILITY STATEMENT Approved for public release; distribution is unlimited						
13. SUPPLEMENTARY NOTES The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government. IRB Protocol Number: N/A						
14. ABSTRACT The Segmented Mirror Telescope (SMT) housed at the Naval Postgraduate School is a unique, state-of-the-art optical instrument built to explore new technologies needed for future space-based telescopes. A discrete Fourier transform wavefront reconstruction technique developed at Lawrence Livermore National Laboratory is discussed in this thesis as applied to a hexagonal aperture. A Fourier domain implementation of a spatial-frequency modal controller for a simple spring-mass model of a deformable mirror surface is provided by this thesis. This technique avoids more difficult time-domain solutions, is computationally efficient and scalable to much larger multi-input, multi-output systems than the SMT.						
15. SUBJECT TERMS Space telescope, Segmented mirror, Adaptive optics, Wavefront reconstruction, Spatial-frequency modal controller, Multi-input Multi-output						
16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT UU	18. NUMBER OF PAGES 69	19a. NAME OF RESPONSIBLE PERSON	
a. REPORT Unclassified	b. ABSTRACT Unclassified	c. THIS PAGE Unclassified			19b. TELEPHONE NUMBER (include area code)	

THIS PAGE INTENTIONALLY LEFT BLANK

Approved for public release; distribution is unlimited

SEGMENTED MIRROR TELESCOPE MODEL AND SIMULATION

Travis W. Axtell
Electronics Engineer
Space and Naval Warfare Systems Center Atlantic
Charleston, South Carolina
B.S., Electrical Engineering, Clemson University, 2007
Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN ELECTRICAL ENGINEERING
from the
NAVAL POSTGRADUATE SCHOOL
June 2011

Author: Travis W. Axtell

Approved by: Roberto Cristi
Thesis Co-Advisor

Jae-Jun Kim
Thesis Co-Advisor

Clark Robertson
Chair, Department of Electrical and Computer Engineering

THIS PAGE INTENTIONALLY LEFT BLANK

ABSTRACT

The Segmented Mirror Telescope (SMT) housed at the Naval Postgraduate School is a unique, state-of-the-art optical instrument built to explore new technologies needed for future space-based telescopes. A discrete Fourier transform wavefront reconstruction technique developed at Lawrence Livermore National Laboratory is discussed in this thesis as applied to a hexagonal aperture. A Fourier domain implementation of a spatial-frequency modal controller for a simple spring-mass model of a deformable mirror surface is provided by this thesis. This technique avoids more difficult time-domain solutions, is computationally efficient and scalable to much larger multi-input, multi-output systems than the SMT.

THIS PAGE INTENTIONALLY LEFT BLANK

Table of Contents

1	Introduction	1
1.1	Purpose	2
1.2	Overview	3
1.3	Research Contributions	4
2	Wavefront Reconstruction	5
2.1	Theory	5
2.2	Square Aperture.	7
2.3	Circular Aperture	9
2.4	Hexagonal Aperture	12
3	Fast Fourier Transform Modeling	15
3.1	Deformable Mirror Plant Model	16
3.2	Deformable Mirror Plant Model in State-Space	18
4	Implementation	21
4.1	Wavefront Reconstruction	21
4.2	Fourier Transform Modeling	23
5	Conclusions	31
5.1	Summary	31
5.2	Additional Work	31
	Appendices	33
A	Equation Set for Circular Aperture	33
B	MATLAB code	35

B.1	ComputeSlopes.m	35
B.2	CheckAdd.m	41
C	Additional Figures	43
	List of References	47
	Initial Distribution List	49

List of Figures

Figure 1.1	SMT segmented mirror orientation.	1
Figure 1.2	Projection of the sample wavefront.	3
Figure 2.1	Square grid of phase points that contribute to gradients in (a) Hudgin geometry and (b) Fried geometry.	6
Figure 2.2	Overhead view of gradients for sample wavefront.	10
Figure 2.3	Circular aperture showing slopes near the boundary.	11
Figure 2.4	SMT wavefront sensor topology for a single segment.	12
Figure 3.1	SMT wavefront sensor and actuator topology for a single segment. . .	15
Figure 3.2	Spring-Mass system model of a deformable mirror.	16
Figure 3.3	Simple block diagram for feed-forward and feedback transfer function.	18
Figure 4.1	Square of phase points for generating equations.	22
Figure 4.2	Wavefront estimated reconstruction comparison.	22
Figure 4.3	Error in the wavefront reconstruction.	23
Figure 4.4	Cleaned up estimated wavefront reconstruction.	24
Figure 4.5	Plant model for deformable mirror.	24
Figure 4.6	Controller model for deformable mirror.	27
Figure 4.7	Simulation model for deformable mirror controller.	28
Figure 4.8	Maximum pixel error across the entire aperture for 25 wavefront updates.	28

Figure 4.9	Maximum error extents on either side of the wavefront across the entire aperture for 25 wavefront updates.	29
Figure C.1	SMT numbered actuator topology for a single segment.	44
Figure C.2	SMT wavefront sensor and actuator topology for a single segment. . .	45

List of Acronyms

DCT-III	discrete Cosine Transform
DFT	discrete Fourier transform
DM	Deformable Mirror
DSFT	discrete spatial Fourier transform
DTT	discrete Triangle Transform
FFT2	two-dimensional fast Fourier transform
JWST	James Webb Space Telescope
NPS	Naval Postgraduate School
NRO	National Reconnaissance Office
SLM	Spatial Light Modulators
SMT	Segmented Mirror Telescope
SRDC	Spacecraft Research and Design Center
WFS	Wavefront Sensor

THIS PAGE INTENTIONALLY LEFT BLANK

Executive Summary

The Segmented Mirror Telescope (SMT) was built to develop new techniques for rapid construction of optical space-based telescopes. This system has been transferred to the Naval Postgraduate School (NPS) and is now hosted by the Spacecraft Research and Design Center (SRDC). As a part of this initiative, the SRDC now is the Adaptive Optics Center of Excellence for National Security and is actively pursuing research focused on the SMT and other optics systems.

The most complex problem of optical systems is constructing large primary mirrors of sufficient optical quality. Traditional monolithic mirrors can take up to four years to construct and test. If the mirror is segmented and split into several smaller mirrors, the process becomes much more rapid. The mirror segments for the SMT were constructed in approximately 18 months. These smaller mirrors can also be combined to form much larger mirrors than could have been built by the traditional monolithic processes.

The trade-off in this new technique is an increased complexity of the system. The monolithic mirror is built to act as a coherent reflector across the entire mirror surface. The larger size also means that it is not as susceptible to damage during the rocket launch into orbit. For the segmented system, the mirrors must be aligned correctly. If this process is not done properly and the mirror height between segments varies too much, the resulting imaging is worse. Phasing of the telescope, the highly sensitive alignment process, must be done to ensure this does not occur.

In addition to the mirror surface being segmented, the rear surface also has surface-parallel piezoelectric actuators. These actuators can correct for mirror imperfections. This arrangement is referred as an active-optics deformable mirror system. This allows for increased variance tolerances in the mirror manufacturing process since the mirror can be corrected when the actuators are energized. This technique lowers costs associated with construction and testing but adds complexity to the control system.

In contrast to the active-optics system, SMT is an adaptive-optics system. The telescope includes a Shack-Hartmann Wavefront Sensor (WFS), which provides information about the quality of the light reaching the camera. The light may lose phase coherency and become less planar. This occurs from either atmospheric disturbances, structural jitter and/or mirror imperfections. Each source of error is summed to provide the total error. The WFS provides the feedback

mechanism for the mirror actuators. A closed control loop would be suited for this situation to minimize wavefront error by correcting the light to have a planar wavefront at the camera detector.

In order to correct the light, an estimate of the wavefront must be formed. An implementation of the Poyneer algorithm is developed in MATLAB to handle other aperture shapes, such as the hexagons used in the SMT. The sampling geometry for the Shack-Hartmann is applied to the algorithm. The algorithm provides the information necessary to be used in a real-time implementation of the wavefront reconstruction such that it could be used in a feedback control loop.

A novel controller design that uses the two-dimensional fast Fourier transform (FFT2) is proposed in this thesis to solve this problem. First, a simple model is developed where a deformable mirror surface is treated as a grid of masses and springs. The actuators have surface normal forces applied to individual masses. The equation to describe this arrangement is a triple convolution in both the space and time domains. Because of the complexity of the coupled equations, the FFT2 is employed. In the frequency domain, the problem transforms into a much simpler arrangement of many uncoupled, independent equations.

After developing the mirror model, we designed the controller model. Instead of the traditional design of one controller for each actuator, one controller for each spatial mode of actuators is used. Each controller then provides actuator commands for every actuator. The inverse transform is performed to recover the specific commands for each actuator.

The mirror model and the controller are implemented in this thesis using MATLAB Simulink. The simulation takes advantage of the powerful discrete filter blocksets to efficiently process the transfer functions in parallel. The mirror aperture size is set to be 256x256 to show the algorithm is scalable to much larger apertures than are currently used by state-of-the-art optical systems.

A closed-loop feedback control simulation is performed with time-varying phase wavefront information. The simulation is run to verify the design of the controller and the wavefront error decrease as the actuators affect the mirror surface. The results are presented in the thesis. This form of a controller can be used on the actual SMT after determining the actual coefficients of the mechanical structure of the system.

The implementation of a discrete Fourier transform (DFT) wavefront reconstruction and de-

formable mirror model and controller is developed in this thesis. These tools will be useful in the future research of the SMT and can be applied to an actual system for experimental results. Any techniques that improve the imaging quality of the SMT can be applied to other telescope systems to ensure this technology continues to progress with the goal of providing higher quality image products for less cost.

THIS PAGE INTENTIONALLY LEFT BLANK

Acknowledgements

I am profoundly grateful for my advisors, Roberto Cristi and Jae-Jun Kim. Without their guidance and immense patience, this work would not have been possible. I also want to thank my colleagues in the Spacecraft Research and Design Center. Working with such a great group of people has been rewarding both on a professional and personal level.

My opportunity to be a student at NPS was due to the generous support of the Office of the Secretary of Defense and my home organization, Space and Naval Warfare Systems Center Atlantic. Without these two organizations working together, I would not have been able to attend this school and follow my dreams.

I have been inspired by my family and friends, whom have all offered me continued encouragement and their love during my education.

This work is dedicated to Kayli and Amara. I am eager to see where your dreams take you.

THIS PAGE INTENTIONALLY LEFT BLANK

CHAPTER 1:

Introduction

Traditional space-based optical telescopes employ a single, large primary mirror structure. This mirror has to be painstakingly manufactured to minimize deformation errors across the surface. The testing may take months to years, depending on the program requirements. These large mirrors are costly to manufacture, test and schedule for spacecraft. The astronomy community has developed a number of technologies that allow for segmented mirrors to replace single, large mirrors. This technology is now being developed by the space-based telescope community for the next generation of telescopes [1].

The Naval Postgraduate School (NPS) hosts a Segmented Mirror Telescope (SMT), originally produced by the National Reconnaissance Office (NRO) as a proof-of-concept experimental spacecraft telescope [2]. It incorporates many aggressive concepts into a single working unit to test and learn about the manufacturing of such an instrument.

The primary mirror is segmented into six equal hexagons arranged in the same manner as a traditional mirror. The segments are paired with each other and have two sets of hinges. In this arrangement, the segmented mirror can be folded up for launch and mechanically unfold and effloresce the segments. This allows for a smaller launch configuration, which increases options for launch vehicles.

In addition to the primary mirror being segmented, it also has actuators across the rear of the

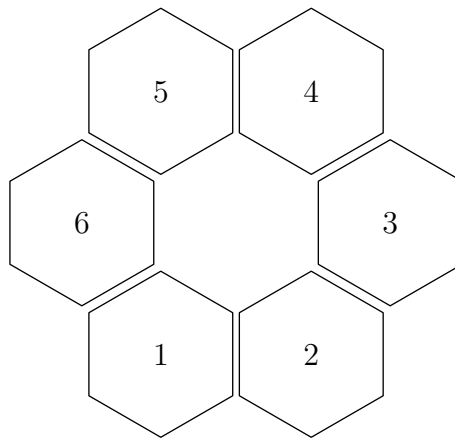


Figure 1.1: SMT segmented mirror orientation.

surface. There are three types of actuators: coarse, fine and face sheet. These actuators can modify the mirror surface in such a way as to correct for abnormalities in the approaching light. The coarse and fine actuators are used as a calibration to give the primary mirror a high-quality optical surface. These actuators make the system an *active-optics system*. With a wavefront sensor these distortions are measured, and a control system adjusts the face-sheet actuators to compensate. In this manner, the telescope can also be considered an *adaptive-optics system*, which is an improvement over active optics. What makes this unique is that the primary mirror has actuators, whereas most adaptive-optics systems make use of a smaller deformable mirror such as a fold mirror that is after the primary in the optical path [3].

The National Security Space Strategy [4] outlines the importance of systems like the SMT. Similar systems offer “unprecedented advantages in national decision-making, military operations and homeland security.” The United States must maintain the benefits afforded by our exquisite systems in the evolving strategic environment. Developing technology that improves the design of the SMT directly supports these goals.

1.1 Purpose

Some of the background fundamental concepts required for a space-based optical telescope that functions as an adaptive-optics system are developed in this thesis. The SMT has a unique geometry that requires finesse in application of these concepts. The concepts are introduced for a simple square mirror and then applied to the SMT and modeled. Simulation results are presented. The improvements made to the SMT model and controller will result in more accurate analysis in future space telescope research. This work benefits government agencies and companies interested in space telescopes.

The purpose of this research is to improve the quality of the imaging results of the telescope. The ideal wavefront is planar at the camera sensor; although, the wavefront can be distorted from the ideal geometry for a number of reasons. Atmospheric mixing and motion can cause non-uniform delays in the wavefront, a common problem for ground-based astronomical telescopes. For spacecraft telescopes, the structural jitter from the motion of changing the satellite pointing direction can deform the optics. For Earth-observation satellites, this is a great concern because the pointing direction is changed often during operation. In this research, a sample wavefront is used in the discussion. The three-dimensional image of the wavefront is shown in Figure 1.2. The red areas indicate the leading wavefront, and the blue areas indicate the lagging wavefront.

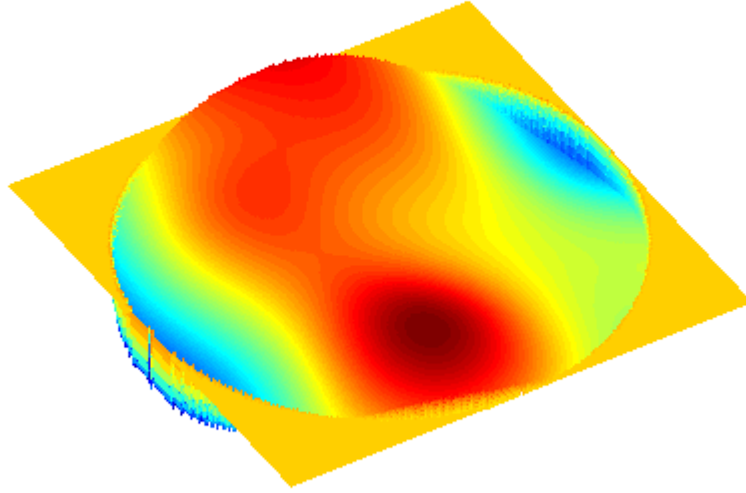


Figure 1.2: Projection of the sample wavefront.

1.2 Overview

The background theory of wavefront reconstruction, which is a process employed to correct the optical system, is provided in Chapter 2. This process is applied to a square and circular apertures, which was developed in previous work [5,6]. Wavefront reconstruction is then discussed for the hexagonal aperture and the unique characteristics of the SMT.

The two-dimensional fast Fourier transform (FFT2) is used to model the SMT in Chapter 3. Use of the FFT2 leads to a computationally efficient algorithm that yields good results. This technique yields decoupled dynamics and a more efficient control design. This improves the scalability of the processing such that larger apertures can be computed without large increases in computational time.

The implementation and simulation work done for this thesis is covered in Chapter 4. Wavefront reconstruction from example generated data is presented. The technique can also be applied to noisy slope data from actual systems. The Deformable Mirror (DM) controller using the discrete

Fourier transform (DFT) is also described, and the technique of building the actual model and its simulation are also presented. The transfer function equivalents are considered to reduce the order of the system. Sample phase data is used to illustrate the performance of the system.

The summary of the work and conclusions with recommended follow on research for the SMT are discussed in Chapter 5.

1.3 Research Contributions

Square and circular apertures are discussed in existing literature and analyzed in great detail [5–8]. Although upcoming telescopes use hexagonal apertures, they are not widely treated in the literature. The required modifications for the wavefront reconstruction algorithm of a circular aperture to correctly model a hexagonal aperture are provided.

Zernike polynomials are used commonly in literature and practice to characterize systems and wavefronts [9]. Although the polynomials are particularly suitable to model optical wavefronts, they are computationally inefficient. An alternate technique that is computationally efficient is desired for realtime, closed-loop operations of a DM. A spatial frequency wavefront reconstruction technique is developed as a viable alternate to Zernike polynomials.

A further contribution is also made in the control system design. In a traditional control design, commands are generated for each actuator individually in the system. This does not scale efficiently as the number of actuators increases. A modal controller is developed to correct for wavefront distortions in the spatial frequency domain.

CHAPTER 2:

Wavefront Reconstruction

2.1 Theory

Wavefront reconstruction is the mathematical process of using the sensed wavefront gradient information and determining an estimated equivalent wavefront. An approaching wavefront has very fine perturbations in the phase which can be observed in the resulting images as distortions. The differences in phase can be specified by using the *optical path length* [10]

$$\delta\phi = \frac{2\pi}{\lambda}n\delta d \quad (2.1)$$

where $\delta\phi$ is the difference in phase, λ is the wavelength of light, n is the *index of refraction* of the media and δd is the minute difference in the physical path length, expressed as a lump sum of all of the distortions. The quantity nd is the optical path length. In the laboratory environment, the medium is air. The optical path length is usually considered for the center wavelength of interest, in this case within the visible light band. Thus, the correction cannot be applied uniformly for all wavelengths, as each has a different optical path length.

These differences reduce the resolution and clarity on the resulting image. To correct for the phase differences, the mirror is properly reformed by a set of actuators. This can be seen in Figure 1.2, where the mirror needs to provide a negative phase adjustment for the red areas by pushing the mirror back and a positive phase adjustment for the blue areas by pushing the mirror forward, so that when the light arrives at the mirror surface, the wavefront reflection is planar. The end result is improved imaging.

In the actual implementation of Equation (2.1), the phase ϕ is an estimate that cannot fully eliminate all errors. The phase estimates $\hat{\phi}(x,y)$ are computed across a two-dimensional grid of values on the basis of observed local gradients. The gradients are sampled at a low spatial frequency across the aperture and, as a consequence of the sampling, can only reconstruct specific low-order modes of the wavefront. These gradients are then processed to determine a wavefront that has local gradients of the same form as the original wave. The device that samples the wavefront is called the wavefront sensor [11]. There are several different wavefront sensors; the one that is discussed in this thesis is the *Shack-Hartmann sensor* [12], which is a grid of

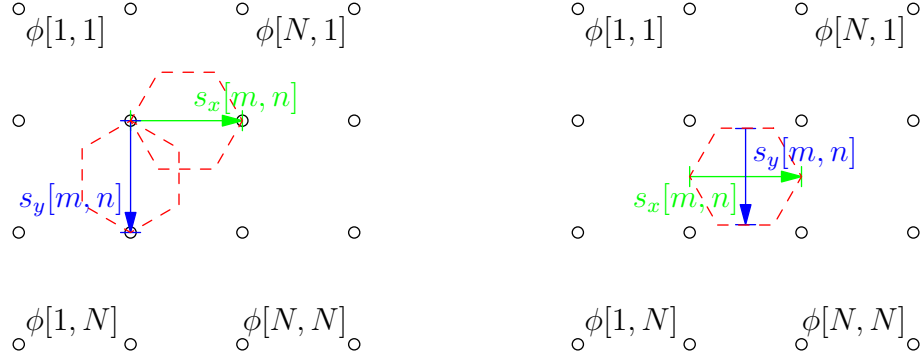


Figure 2.1: Square grid of phase points that contribute to gradients in (a) Hudgin geometry and (b) Fried geometry. Shack-Hartmann sensors are shown in red dashed lines with hexagon lenslets.

small lenses that are side-by-side. Each lenslet focuses the received light to a point on the focal plane of a secondary camera. If the point does not appear at the center of the lenslet optical axis on the camera focal plane, then the light has wavefront variation. The position of these points relative to their expected centered position yields the information on the phase gradient.

There are two common sensor geometries developed by Hudgin [7] and Fried [8] shown in Figure 2.1. These geometries vary by the location of the Wavefront Sensor (WFS) relative to the *phase points*, which are abstract locations in the optical plane that have the equivalent gradients between adjacent phase points as registered by the sensors. The Hudgin geometry is easier to work with initially in simulation but has the drawback that the sensor's lenslets have overlap, which is not commonly done in actual implementation. Fried geometry is more commonly used to model the actual Shack-Hartmann sensors. The SMT uses hexagon-shaped lenslets for its sensors. These pack together densely without overlap and focus all available light on the sensor detection focal plane.

Wavefronts can be reconstructed using a variety of basis functions. Some reconstructions use Zernike polynomials, which are particularly suitable to describe the common optical characterizations of astigmatism, coma, defocus and others [9]. Although a full expansion is computationally inefficient, only a few terms are needed to characterize a wavefront.

An alternate decomposition by the DFT yields the desirable property of being computationally efficient and still only providing a few dominant coefficients [6]. The work in this thesis uses this technique.

2.2 Square Aperture

The square aperture is a primitive geometry that allows for a simplified reconstruction algorithm. The design assumes that the actuators and phase points are co-located. Thus, the actuators directly influence the phase points (and, thereby, the gradients). For a $N \times N$ grid of phase points $\phi[m, n]$, Hudgin's geometry model defines the gradients as the difference of the neighbor phase points:

$$s_x[m, n] = \phi[m + 1, n] - \phi[m, n] \quad (2.2a)$$

$$s_y[m, n] = \phi[m, n + 1] - \phi[m, n]. \quad (2.2b)$$

Alternatively, Fried's model defines the gradients as the average of the neighbor phase points:

$$s_x[m, n] = \frac{1}{2}(\phi[m + 1, n] - \phi[m, n] + \phi[m + 1, n + 1] - \phi[m, n + 1]) \quad (2.3a)$$

$$s_y[m, n] = \frac{1}{2}(\phi[m, n + 1] - \phi[m, n] + \phi[m + 1, n + 1] - \phi[m + 1, n]). \quad (2.3b)$$

In either case, the result is two sets of slope data that are an $(N - 1) \times (N - 1)$ grid of values, rather than $N \times N$ because the slopes can only be computed between data points. Slopes are defined as the local gradients separated into the individual components, in this case s_x and s_y . Using the notion that any closed path along the mirror surface must result in a net slope change of zero, Freischlad determined the last index of values along the boundaries as [5]:

$$s_x[N, n] = - \sum_{k=1}^{N-1} s_x[k, n] \quad (2.4a)$$

$$s_y[m, N] = - \sum_{k=1}^{N-1} s_y[m, k]. \quad (2.4b)$$

The two sets of slope data are now $N \times N$. An alternate method is to assume that $\phi[m, n]$ is periodic and that $\phi[1, N + 1] = \phi[1, 1]$ and so on. In this method, the last slopes can be computed, also resulting in $N \times N$ data. This periodic assumption is included in the definition of the DFT.

After s_x and s_y are determined from the sensors, the reconstruction of the $\hat{\phi}$ can be completed using the FFT2. The first step is to take the FFT2 of the s_x and s_y data, as

$$S_x[k, l] = \sum_{p=0}^{N-1} \sum_{q=0}^{N-1} s_x[p, q] e^{-\frac{j2\pi}{N}(kq+lp)} \quad (2.5a)$$

$$S_y[k, l] = \sum_{p=0}^{N-1} \sum_{q=0}^{N-1} s_y[p, q] e^{-\frac{j2\pi}{N}(kq+lp)}. \quad (2.5b)$$

The data can be filtered in the frequency domain. This filtering is equivalent to solving Equations (2.3a) and (2.3b) for ϕ in the frequency domain to obtain

$$\hat{\Phi}[k, l] = \begin{cases} 0, & \text{if } k, l = 0 \\ \frac{[e^{-\frac{j2\pi k}{N}} - 1]S_x[k, l] + [e^{-\frac{j2\pi l}{N}} - 1]S_y[k, l]}{[4(\sin^2(\frac{\pi k}{N}) + \sin^2(\frac{\pi l}{N}))]}, & \text{otherwise} \end{cases} \quad (2.6a)$$

$$\hat{\Phi}[k, l] = \begin{cases} 0, & \text{if } k, l = 0; k, l = N/2 \\ \frac{[e^{-\frac{j2\pi k}{N}} - 1][e^{-\frac{j2\pi l}{N}} + 1]S_x[k, l] + [e^{-\frac{j2\pi l}{N}} - 1][e^{-\frac{j2\pi k}{N}} + 1]S_y[k, l]}{[8(\sin^2(\frac{\pi k}{N})\cos^2(\frac{\pi l}{N}) + \sin^2(\frac{\pi l}{N})\cos^2(\frac{\pi k}{N}))]}, & \text{otherwise.} \end{cases} \quad (2.6b)$$

Equation (2.6a) is valid for Hudgin geometry, while Equation (2.6b) is valid for Fried geometry.

The resulting wavefront phase estimate is available after taking the inverse FFT2:

$$\hat{\phi}[m, n] = \frac{1}{N^2} \sum_{p=0}^{N-1} \sum_{q=0}^{N-1} \hat{\Phi}[p, q] e^{\frac{j2\pi}{N}(mp+nq)}. \quad (2.7)$$

The phase value at the phase points is now known. This information can then be processed by the controller to properly deform the mirror surface and improve the resulting image. Since the traditional deformable mirror has the actuators co-located with the phase points, the controller can easily make the necessary changes that directly improves the imaging results.

2.2.1 Boundary Conditions

The combination of the sums in Equations (2.4a) and (2.4b) imply that there are artifacts along the boundary of two of the four sides. The sums of noisy data points result in a larger variance of the noise on the sum. This is an undesirable consequence of this technique.

The boundary is where the majority of the error occurs when computing the wavefront. These errors can result in large discontinuities which affect the reconstruction algorithm at other points across the entire aperture. This can be explained since the impulse response of the algorithm is shift variant, although the reconstruction is unbiased [5]. That is to say, the sums in Equations (2.4a) and (2.4b) have detrimental effects across the entire aperture in reconstruction; a better technique must address this issue.

2.3 Circular Aperture

The circular aperture was analyzed by Poyneer [6]. Wavefront reconstruction in the center of the aperture is relatively straightforward from the square aperture discussion in Section 2.2. However, along the circular boundary, there are large errors in the slopes as calculated by Equation (2.3). These occur because the area external to the circular aperture is set to zero in the matrix grid of data. The slope data is then calculated incorrectly, and the error propagates into the reconstructed wavefront. Several modifications are needed to understand how the slopes are affected by the boundary.

In Figure 2.2, the gradients are shown as scaled vectors sampled at a very low rate. The vectors near the boundary contain large errors, as can be seen by the larger magnitude and even opposite phase. Additionally, some of the gradients are non-zero outside of the aperture.

2.3.1 Boundary Conditions

The definition of the slopes in Equations (2.2) has to be altered to accommodate the circular boundary. At each boundary point $[m, n]$, let

$$s_x[m, n] = i_x[m, n] + b_x[m, n] \quad (2.8a)$$

$$s_y[m, n] = i_y[m, n] + b_y[m, n] \quad (2.8b)$$

where the values $i_x[m, n]$ and $i_y[m, n]$ are internal slopes near the boundary that are measured using Equations (2.2); these are the necessary values to compute wave-front phase reconstruction correctly. Additionally, the $b_x[m, n]$ and $b_y[m, n]$ are the boundary slopes that are incorrect; removal of these terms is necessary for proper wavefront reconstruction.

The relationship between the internal and boundary gradients are defined by the matrix

$$\mathbf{M}\underline{u} = \underline{c} \quad (2.9)$$

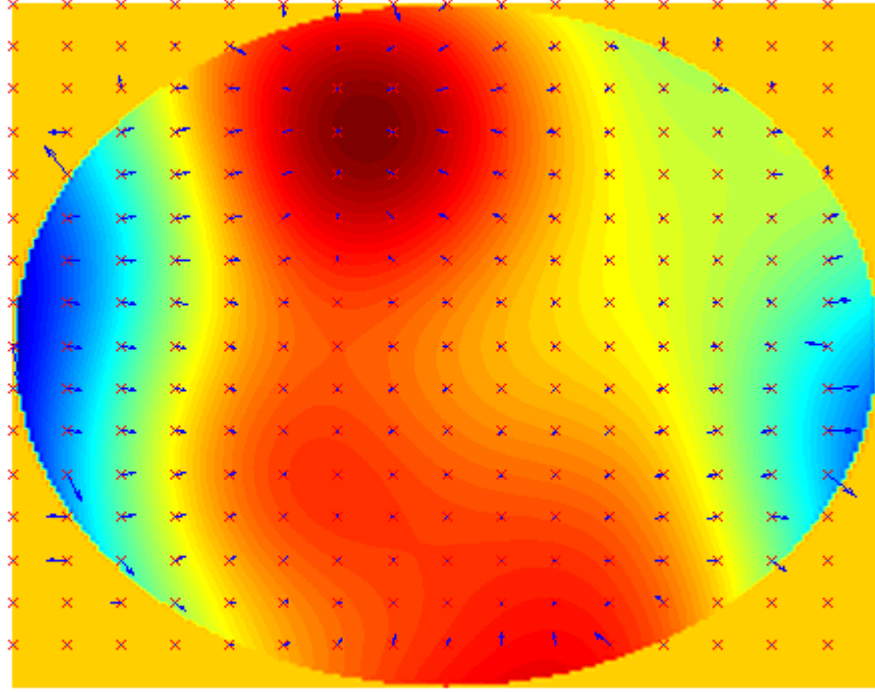


Figure 2.2: Overhead view of gradients for sample wavefront.

where \mathbf{M} is a matrix determined by the geometry, \underline{u} is a vector of unknown boundary slopes and \underline{c} is a vector of the measured slopes which slopes cross the boundary.

The matrix \mathbf{M} is developed from the equations that relate the entries of \underline{u} to \underline{c} . To solve for \underline{u} , the inverse of \mathbf{M} is required. Equation (2.9) has an infinite number of solutions since it does not solve for the DC term (colloquially known as the *piston*). This occurs because, while the slopes of the phase are known, the actual phase is not. Since \mathbf{M} is singular, \mathbf{M} might be better conditioned by singular value decomposition while keeping the dominant singular values.

In Figure 2.3, the unknown boundary slopes are labeled sequentially in u , while the known internal slopes are $s_x[m, n]$ and $s_y[m, n]$. The u slopes all cross the circular boundary edge and is shown as a dotted line. The zero vectors shown indicate that the two connected points are both external to the circular aperture.

By writing the mesh equations for small square edges between adjacent phase points, we obtain

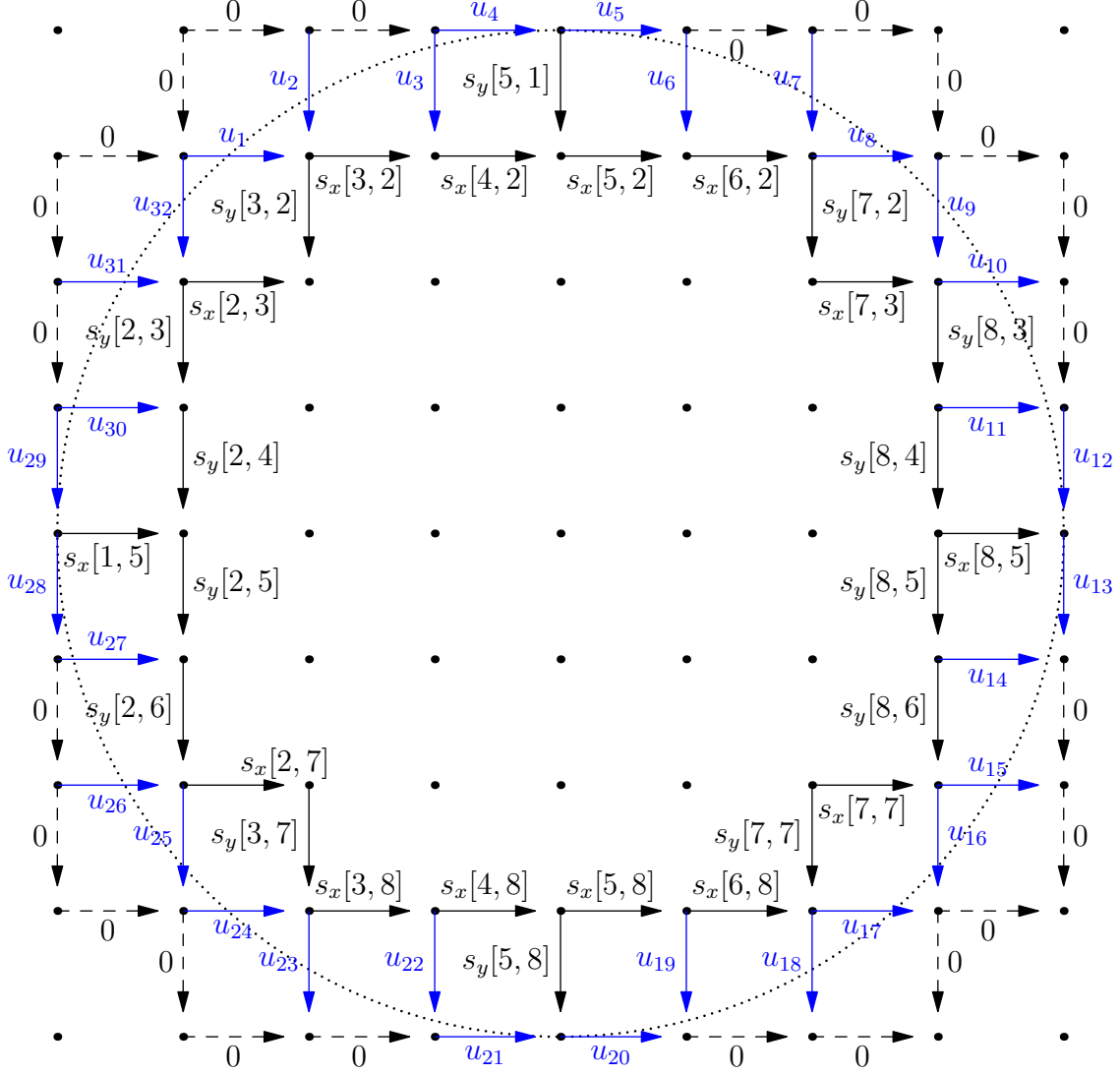


Figure 2.3: Circular aperture showing slopes near the boundary.

a set of linear equations. This method generates the linear combinations necessary to solve for the non-noisy data exact solution or the noisy data least-squares solution. These equations directly relate the measured slopes to the unknowns and zero external slopes. A starting location must be selected along the boundary, and then a path is traced along the boundary until the aperture edge is closed. Depending on the geometry, we see that the \underline{c} entry is a linear combination of some $s_x[m,n]$ and $s_y[m,n]$. Upon solving for the \underline{u} solution, we obtain each resulting \underline{u} entry as either a $b_x[m,n]$ or $b_y[m,n]$. Knowing the $b_x[m,n]$ or $b_y[m,n]$ information, we solve Equation (2.8) for the needed $i_x[m,n]$ and $i_y[m,n]$, which is trivial. The results correct the slope data along the boundary, and the data is now used for reconstruction of the phase ϕ .

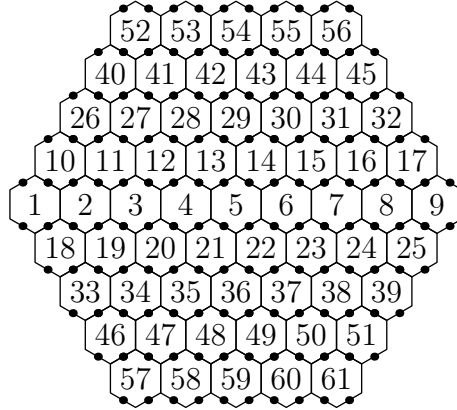


Figure 2.4: SMT wavefront sensor topology for a single segment with modified Fried geometry phase point locations.

2.4 Hexagonal Aperture

The hexagonal aperture is a modification of a circular aperture. It is more desirable for multiple mirror systems than the latter since its segments can easily fit together in a continuous fashion. It has been commonly used on many recent space telescopes, including the SMT and the James Webb Space Telescope (JWST).

The linear edges make the mirror segments easier to align with one another, such as by using edge-detection sensors. These sensors are able to detect the neighbor segment mirror heights and adjust the coarse and fine actuators to place the heights of the mirror surfaces very closely to one another at the nanometer scale. This calibration is referred to as phasing the telescope and is seldom needed to be repeated for on-orbit operation but is performed often for the ground-based telescopes.

For the SMT, the 61 WFS per segment pictured in Figure 2.4 are packed together densely. However, their combined edge does not result in an exact hexagonal shape along the outer edge. The phase points should be placed along the edges of the individual lenslets. This makes the grid uniform in rectangular coordinates. This is a more convenient arrangement than the one originally depicted in Figure 2.1. If the phase points were completely inside the lenslet for each lenslet, there would be no information about the relationship between the phase points of adjacent lenslets. Likewise, if the phase points were all external to the lenslet (as was shown in Figure 2.1), there would be an incorrect relationship between phase points and the measured slope data due to an overlap between the neighbor lenslet phase points. This arrangement would indicate that the phase in one lenslet is influenced by the neighbor, which is not the case. Placing

the phase points along the edge in this manner solves these two issues. The geometric implications of how these hexagon lenslets focus different amounts of light per axis on the sensor detector warrant additional inquiry. The distance between two neighbor phase points across a row is different than the distance between two neighbors down a column. This information must be included in the phase point calculation, or the visible result will be distorted.

In the phase point arrangement depicted in Figure 2.4, a single segment of the mirror can be stored in a rectangular grid of 10 rows and 18 columns. If all six segments were in a single matrix, it can be done in as small of a matrix as 30 rows and 54 columns.

2.4.1 Boundary Conditions

The important consequence of the rectangular grid phase point arrangement is that the boundary correction algorithm for the circular aperture can now be directly applied to the hexagon. Without this relationship, the formulas must be modified or the slopes considered to lie along the vectors that compose the hexagonal lattice defined by the center points of the 61 WFS. The actual formulation of the equations depends on how the aperture boundary crosses the phase points. Every possible phase point arrangement should be considered; in a rectangular grid arrangement of four phase points, there are 16 possible combinations of which phase points are inside and which are outside of the aperture boundary; since each of these is handled in exactly the same way as they would for the circular aperture, no modification is required.

THIS PAGE INTENTIONALLY LEFT BLANK

CHAPTER 3:

Fast Fourier Transform Modeling

The FFT2 is an implementation of the DFT that is computationally efficient. In a two-dimensional grid of $\hat{\phi}[m, n]$, the two-dimensional DFT provides the spatial frequency content of the wavefront, showing that the dominant modes of the wavefront are lower frequencies [13]. Therefore, in practice only the lower order modes are needed to correct the largest error terms.

Breaking the wavefront down into spatial modes is well suited for the actuators. On the SMT, the actuators are placed into a triangle mesh along the structural supports on the rear surface of the mirror. This truss lattice is independent from the hexagon lattice of the WFS. This geometric relationship between the phase points and the actuators varies across the entire segment as seen in Figures 3.1, C.1 and C.2. There are two methods for dealing with these different lattices. The first is a coordinate transformation. The second method is working in the spatial frequency domain.

In the traditional DM controller, the phase points and actuators are co-located. This makes for a simple controller because the wavefront phase error is known at the locations of the actuators. Thus, a simple actuation change causes a wavefront change, with the goal of being an improvement in the resulting image. For the SMT geometry, this is not the case. The wavefront phase is sampled along the hexagon lattice; the actuators are found along the truss lattice.

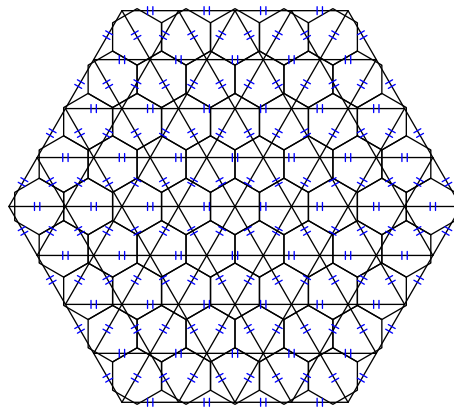


Figure 3.1: SMT wavefront sensor and actuator topology for a single segment.

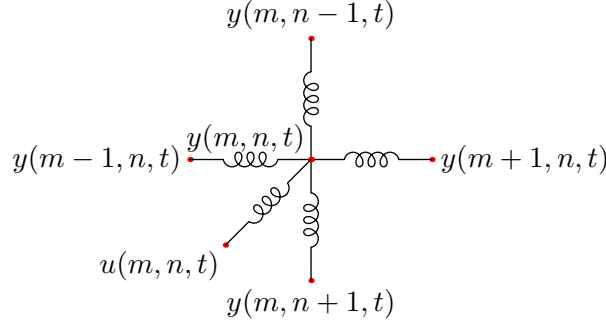


Figure 3.2: Spring-Mass system model of a deformable mirror.

3.1 Deformable Mirror Plant Model

The plant model mathematically represents the physical system, in this case, the DM position. To simplify the analysis, the mirror is thought of as a system of masses connected to one another with springs. Each mass is connected to four neighbors, as shown in Figure 3.2. In this manner, the mirror surface forms a large Cartesian grid of mass points. Each point also has a fifth spring connection that is the height displacement out of the plane of the mirror.

From Figure 3.2, utilizing the concept of Hooke's law, we can relate the vertical displacement $y(m, n, t)$, velocity $\dot{y}(m, n, t)$ and acceleration $\ddot{y}(m, n, t)$ at location (m, n) and time t to the actuators $u(m, n, t)$ and the neighbor's displacement as

$$\begin{aligned}
\ddot{y}(m, n, t) = & -\alpha_0 \dot{y}(m, n, t) - \beta_0 y(m, n, t) + \gamma_0 u(m, n, t) \\
& - (\alpha_{0,1} (\dot{y}(m, n, t) - \dot{y}(m, n-1, t))) \\
& - (\alpha_{0,-1} (\dot{y}(m, n, t) - \dot{y}(m, n+1, t))) \\
& - (\alpha_{1,0} (\dot{y}(m, n, t) - \dot{y}(m-1, n, t))) \\
& - (\alpha_{-1,0} (\dot{y}(m, n, t) - \dot{y}(m+1, n, t))) \\
& - (\beta_{0,1} (y(m, n, t) - y(m, n-1, t))) \\
& - (\beta_{0,-1} (y(m, n, t) - y(m, n+1, t))) \\
& - (\beta_{1,0} (y(m, n, t) - y(m-1, n, t))) \\
& - (\beta_{-1,0} (y(m, n, t) - y(m+1, n, t)))
\end{aligned} \tag{3.1}$$

where the α coefficients represent damping and the β coefficients represent the spring effect

from the neighbors. The negative signs are to ensure the positive α and β values result in a stable system.

The transfer function from the input $u(m, n, t)$ to the output $y(m, n, t)$ is defined as

$$G(m, n, s) = \frac{\gamma_0}{s^2 + \sigma s + \rho}, \quad (3.2)$$

and the transfer function between the center mass and its neighbors are of the form

$$H(m, n, s) = \frac{-\alpha(m, n)s - \beta(m, n)}{s^2 + \sigma s + \rho} \quad (3.3)$$

with $m, n = -1, 0, 1$. We also define $\sigma = \alpha_0 + \alpha_{0,1} + \alpha_{0,-1} + \alpha_{1,0} + \alpha_{-1,0}$ and $\rho = \beta_0 + \beta_{0,1} + \beta_{0,-1} + \beta_{1,0} + \beta_{-1,0}$. Since σ and ρ are always positive, this system is stable. If α and β do not vary with the index, the result is a symmetric output where the energy dissipates into the neighbors evenly. These values affect the pole locations of the denominator, and this determines the response time of the system. In this manner, this abstract model can be applied to a DM if these values can be determined from either structural analysis or testing.

Equation (3.1) can be expressed in the time domain as

$$y(m, n, t) = \int_{-\infty}^{\infty} \left[\sum_{l_1=-1}^1 \sum_{l_2=-1}^1 [h(l_1, l_2, t)y(m-l_1, n-l_2, t-\tau)] + g(t)u(m, n, t-\tau) \right] d\tau \quad (3.4)$$

which shows a triple convolution in space and time. In the time-domain analysis, all neighbor actuators influence each other. The solution must be obtained by concurrently solving the entire set of equations for the entire mirror surface. Equation (3.4) can be written in the Laplace domain as

$$Y(m, n, s) = \sum_{l_1=-1}^1 \sum_{l_2=-1}^1 [H(l_1, l_2, s)Y(m-l_1, n-l_2, s)] + G(m, n, s)U(m, n, s) \quad (3.5)$$

where $G(m, n, s)$ and $H(m, n, s)$ are given in Equations (3.2) and (3.3). For every s , define the two-dimensional $N \times N$ FFT2 as

$$Y(k_1, k_2, s) = \sum_{m=0}^{N-1} \sum_{n=0}^{N-1} Y(m, n, s) e^{\frac{-j2\pi}{N}(k_1 m + k_2 n)} \quad (3.6)$$

and

$$H(k_1, k_2, s) = \sum_{m=0}^{N-1} \sum_{n=0}^{N-1} H(m, n, s) e^{\frac{-j2\pi}{N}(k_1 m + k_2 n)} \quad (3.7)$$

with $k_1, k_2 = 0, \dots, N-1$. Then the discrete spatial convolution in Equation (3.5) becomes

$$Y(k_1, k_2, s) = H(k_1, k_2, s)Y(k_1, k_2, s) + G(s)U(k_1, k_2, s). \quad (3.8)$$

This combination of transforms greatly simplifies a three-dimensional convolution into multiplication in the frequency domain. In the frequency domain, each frequency is treated independently from the others; the equations can be solved individually with much less computation required.

3.2 Deformable Mirror Plant Model in State-Space

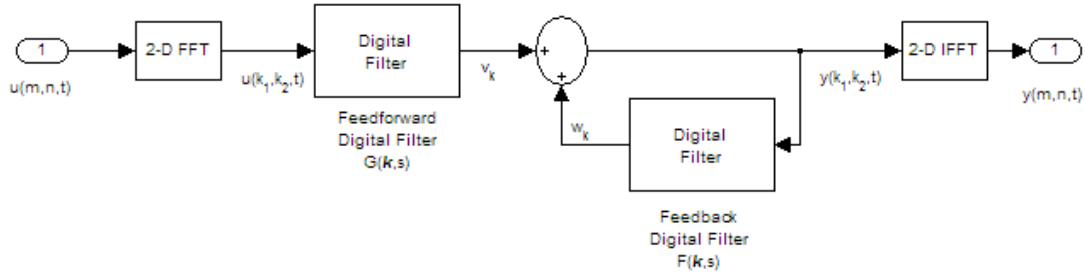


Figure 3.3: Simple block diagram for feed-forward and feedback transfer function.

The purpose of the mirror controller is to determine the actuator command values at each point (m, n) across the entire mirror, $u(m, n, t)$ to remove as much wavefront error as possible. In an active-optics system, the actuators might be moved once during a calibration such that the mirror form is as accurate as possible, correcting the radius of curvature to match designed radius. In an adaptive-optics system, the actuators are used as a function of time to correct for the varying wavefront of light. It is possible to attempt to do both of these corrections concurrently, but there would be a loss of control authority in that some actuators will have a limited range of

motion since the actuators are already using their stroke length for correcting the radius of the mirror surface.

In order to design the controller, first a mathematical model in terms of the state-space equations is derived for the simple block diagram of transfer functions shown in Figure 3.3. The general relationship between a transfer function and its state-space equations is

$$G(s) = \frac{N_0(s)}{D_0(s)} = C(sI - A)^{-1}B + D \quad (3.9)$$

where $N_0(s)$ is the numerator polynomial and $D_0(s)$ is the denominator polynomial of the transfer function, and A , B , C , D are the matrices that relate the inputs, outputs, and states. In the following, the D matrix is removed because it is not used for the model.

For the feed-forward and feedback blocks, the state-space equations can be derived as

$$\dot{\underline{z}}_k = A_0 \underline{z}_k + b_0 u_k, \quad (3.10a)$$

$$v_k = c_0 \underline{z}_k, \quad (3.10b)$$

$$\dot{\underline{s}}_k = A \underline{s}_k + b y_k, \quad (3.10c)$$

$$w_k = c_k \underline{s}_k, \quad (3.10d)$$

and

$$y_k = v_k + w_k. \quad (3.11)$$

where \underline{z}_k and \underline{s}_k are the internal states of the systems in Figure 3.3. In Equation (3.10), A_0 relates the neighbor masses to one another, b_0 relates the actuators to the masses and c_0 relates the internal state to the output. Likewise, the coefficients A , b and c_k do the same for the internal state that varies in spatial frequency. The k subscript indicates that these equations and coefficients must be applied to each spatial frequency vector $\langle k_1, k_2 \rangle$ individually in actual implementation. These equations can be combined to a single state-space model to get

$$\underline{x}_k = \begin{bmatrix} \underline{z}_k \\ \underline{s}_k \end{bmatrix}, \quad (3.12a)$$

$$\dot{\underline{x}}_k = \underbrace{\begin{bmatrix} A_0 & \underline{0} \\ bc_0 & A + bc_k \end{bmatrix}}_{=f_k} \underline{x}_k + \underbrace{\begin{bmatrix} b_0 \\ 0 \end{bmatrix}}_{=\underline{g}} u_k, \quad (3.12b)$$

$$(3.12c)$$

and

$$y_k = \underbrace{\begin{bmatrix} c_0 & c_k \end{bmatrix}}_{=h_k} \underline{x}_k \quad (3.13)$$

where f_k , \underline{g} and h_k are the equivalent A , B and C general matrices, respectively. The zero vector, $\underline{0}$, indicates its row dimension will match A_0 to correctly fill the matrix.

This state-space model is at the basis of the overall controller we will develop in the next chapter. What is significant about this approach is that the models represented by Equations (3.12) and (3.13) are all decoupled and can be individually controlled.

CHAPTER 4:

Implementation

In this chapter, we test the estimation and control techniques developed using simulations in MATLAB and Simulink software. Example phase data sets were generated by algorithms developed by Melissa Corley [14] while with the Spacecraft Research and Design Center (SRDC) at NPS. Her code has excerpts from Chris Wilcox on Spatial Light Modulators (SLM) [15]. The simulated phase data size of the aperture was 1024x1024 phase points, much larger than existing adaptive-optics systems at the SRDC. This large dimension was chosen to show the scalability of the algorithm.

4.1 Wavefront Reconstruction

The circular aperture solution forms the basis for solving a number of different geometries. Poyneer outlined the general theory of how to accomplish this [6]. In this section, a particular algorithm for this technique is developed.

One of the difficulties of applying a Fourier-based reconstruction algorithm is the fact that apertures are generally not square but circular or polygonal, which does not fit with the square grid of a FFT2. In order to overcome this difficulty, we need to make some assumptions on the data outside the aperture and within the grid of the FFT2. In general, we assume the extra values to be zero. Particular care has to be taken at the boundary of the aperture since the approach developed by Poyneer leads to algorithms which are fairly sensitive to boundary conditions [6].

Apertures with various characteristics of *vignetting*, such as mechanical obstructions to the optical pathway, can also be included in the final solution. This can be useful for space-based telescope apertures since the secondary mirror usually has mechanical vignetting from its structural support that holds the mirror in place.

To implement this algorithm, the entire grid of data is iterated to identify each group of four neighbor phase points. The only groups of consequence are along the boundary. The algorithm builds a list of boundary slopes \underline{u} that needs to be solved. If another new unknown is found along the boundary, it is appended sequentially in a list. If it is already in the list, the algorithm uses the number previously assigned. For the square, any slopes that are completely internal to the aperture need to be used in computing the constant for \underline{c} . After the entire grid is processed,

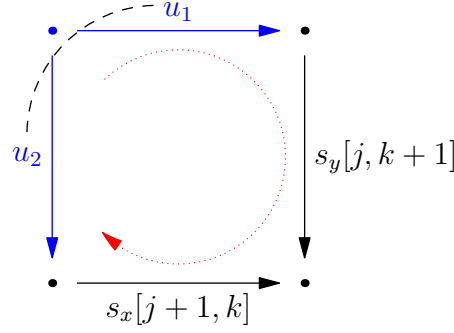


Figure 4.1: Square of phase points for generating equations.

all of the equations are identified. The matrix \mathbf{M} in Equation (2.9) is quickly produced. All of the required information for Equation (2.9) is available to solve the boundary problem. This design allows for a variety of apertures to be solved, including the hexagon.

In Figure 4.1, the black dashed line shows the aperture boundary. The blue phase point dot is outside of the aperture, while the black dots are inside. The red dotted line indicates the mesh equation direction. The unknowns and internal slopes are identified. In this example, the entry in \underline{c} in Equation (2.9) is equal to $s_y[j, k+1] - s_x[j+1, k]$.

After correctly setting up Equation (2.9) variables \mathbf{M} and \underline{c} , \underline{u} is easily solved. The values \underline{u} contains apply to specific $s_x[j, k]$ or $s_y[j, k]$ that are identified by the unknowns. Each \underline{u} value is iterated and then applied to the slopes in Equation (2.8). The results are matrices of $i_x[j, k]$ and $i_y[j, k]$ which are used in the reconstruction algorithm. This technique is applied to the large example data set shown in Figure 1.2. The reconstructed wavefront is shown in Figure 4.2. The reconstruction looks comparable to the original phase data, but there are some artifacts in the external aperture area.

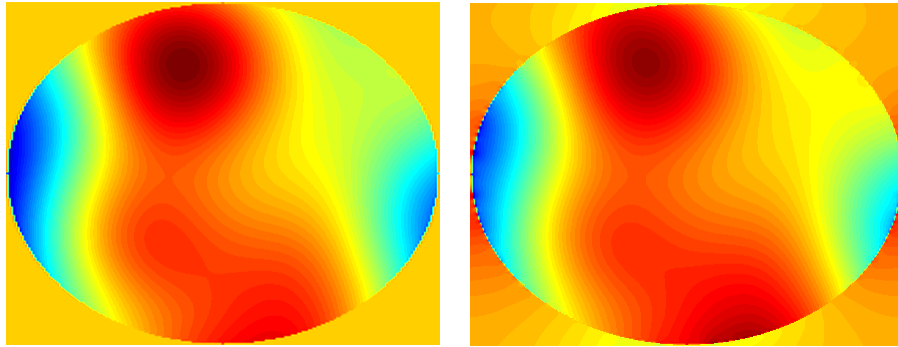


Figure 4.2: (a) Original wavefront; (b) Reconstructed wavefront.

The difference in the reconstructed wavefront from the original (known) wavefront is computed and shown in Figure 4.3. This shows that the larger errors are at the boundary as expected or at the external edge of the boundary. This is a consequence of the filtering, discarding the imaginary components and round-off errors in floating point operations. Since the algorithm is insensitive to the mean value, the comparison is made between the reconstructed data and the mean-subtracted original data.

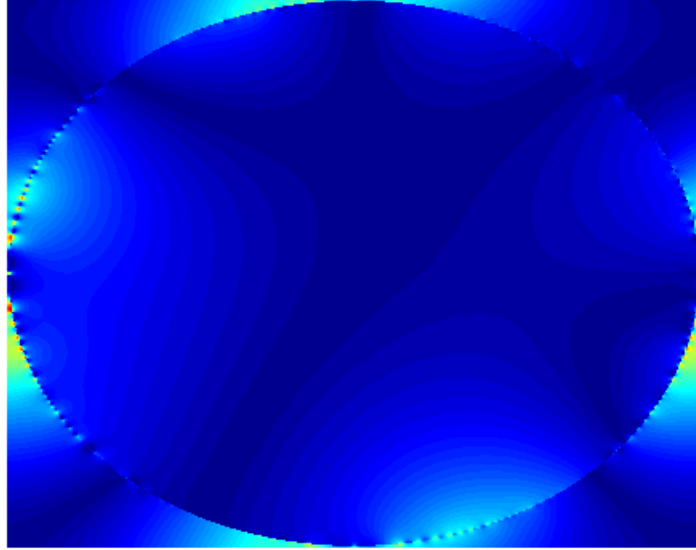


Figure 4.3: Error in the wavefront reconstruction.

To better compare the results, the reconstructed data is further processed by zeroing out any entries outside of the known aperture. In the actual implementation, these values are not used or needed. As can be seen in Figure 4.4, the estimate data closely matches the original data. These additional steps are not necessary in actual implementation and are only included here to help the reader see that the reconstruction was successful.

4.2 Fourier Transform Modeling

The DM is modelled using the discrete spatial Fourier transform (DSFT) to decompose the spatial modes of the mirror. Mathematically, this is done using rectangular matrices. Matrix sizes are padded with zeros to a length that is a multiple of two. This is done so that the FFT2 algorithm can execute efficiently. Previous work has been done using 64x64 matrices [6]. In this thesis, the matrices used were 256x256 in size to show that the algorithm is scalable to the larger apertures that are expected in the future.

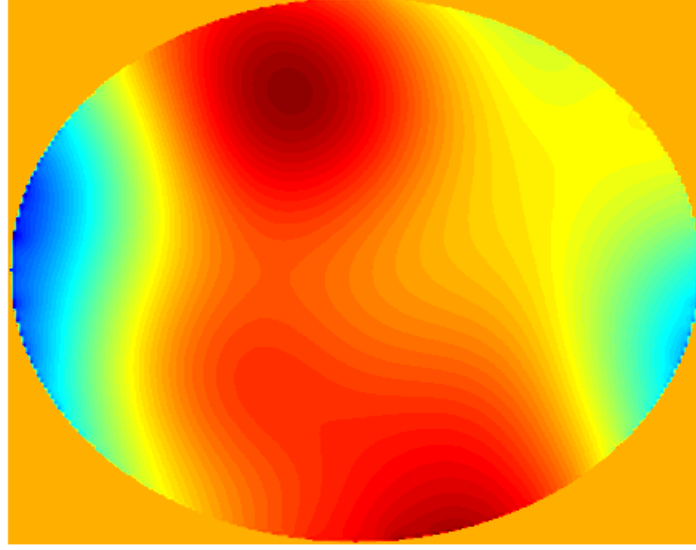


Figure 4.4: Cleaned up estimated wavefront reconstruction.

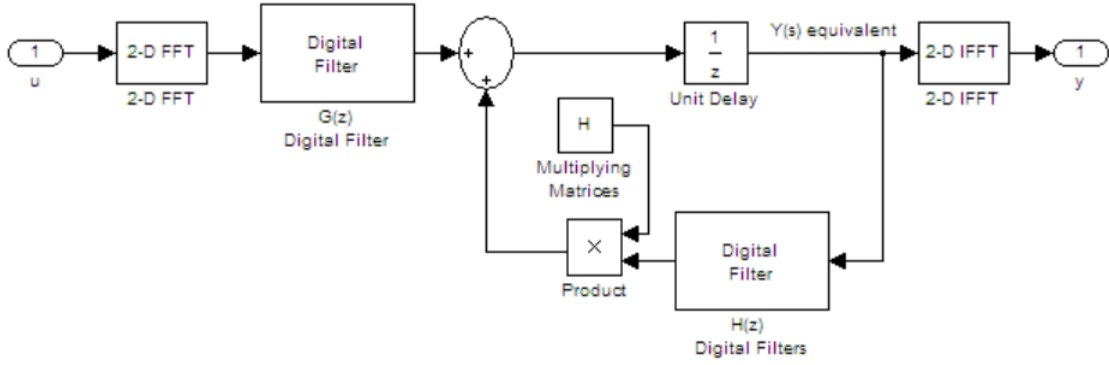


Figure 4.5: Plant model for deformable mirror.

The plant implementation can be seen in Figure 4.5. In order to make the implementation more efficient, the analog transfer functions in s are implemented in discrete time in z by converting from continuous time to discrete time using a zero-order hold (ZOH) over an assigned sampling period T_s . The sampling time can be adjusted to match the physical system response. In order to avoid algebraic loops, we implement the transfer functions to be strictly proper by adding a time delay z^{-1} .

The forward gain in Equation (3.2) is implemented as a scalar multiplier of each spatial frequency. To implement Equation (3.3), two matrices are created, h_α and h_β . Each matrix size is set to be the same as the size of the aperture and has the form

$$h_\alpha(m, n, s) = \begin{bmatrix} 0 & -\alpha_{0,1} & 0 & \dots & 0 & -\alpha_{0,-1} \\ -\alpha_{1,0} & 0 & 0 & \dots & 0 & 0 \\ 0 & 0 & 0 & \ddots & 0 & 0 \\ -\alpha_{-1,0} & 0 & 0 & \dots & 0 & 0 \end{bmatrix}. \quad (4.1)$$

After creating these matrices, their two-dimensional FFTs are computed. The two matrices h_α and h_β are needed for the Laplace s term and the constant term. These matrices are constructed with periodicity. The term that relates the mirror mass to its northern neighbor is wrapped around to the bottom row of the first column; likewise, the term that relates the mirror mass to its western neighbor is wrapped around to the rightmost entry in the first row. The DSFT is applied, and the results are identified as matrices H_α and H_β . The three matrices are used in conjunction with the digital filters as an element-wise matrix multiplier, which implements Equation (3.8) efficiently as compared to Equation (3.4). These matrices are not converted to discrete time, as the discrete filters implement the continuous-time equation.

The purpose of the controller is to meet the design goal of Equation (4.2) given that $e(m, n, t)$ is the error of the mirror state minus the sensed wavefront. The ability of the system to keep the error below this threshold depends upon how rapidly the mirror can respond to the wavefront's constantly evolving phase progression in time. This requirement is expressed by

$$e(m, n, t) = y(m, n, t) - \phi(m, n, t) \quad (4.2a)$$

$$|e(m, n, t)| < \varepsilon, \forall t \geq t_0. \quad (4.2b)$$

The last step is to add an integrator state to the final output of the controller. An integrator controller has additional robustness compared to classical controllers. The integral controller is well suited for tracking of constant reference inputs, which in the case of slowly-varying wavefront mode coefficients is an adequate approximation. The resulting state-space equations are

$$\begin{bmatrix} \dot{u}_k \\ \dot{x}_k \end{bmatrix} = \underbrace{\begin{bmatrix} 0 & 0 \\ \underline{g} & f_k \end{bmatrix}}_{=\bar{f}_k} \begin{bmatrix} u_k \\ x_k \end{bmatrix} + \underbrace{\begin{bmatrix} 1 \\ \underline{0} \end{bmatrix}}_{=\bar{g}} \bar{u}_k \quad (4.3)$$

and

$$y_k = \underbrace{\begin{bmatrix} 0 & h_k \end{bmatrix}}_{=\bar{h}_k} \begin{bmatrix} u_k \\ x_k \end{bmatrix}. \quad (4.4)$$

For each spatial frequency, the matrix dimensions for \bar{f}_k , \bar{g} and \bar{h}_k are 5x5, 5x1 and 1x5, respectively. However, this fifth order system is not of full rank. An uncontrollable mode exists because of a pole/zero cancellation in the transfer function. Referring to Figure 3.3, we obtain the transfer function which shows this cancellation:

$$\frac{Y_k(s)}{U_k(s)} = \frac{N_0(s)}{D(s)} \frac{1}{1 - \frac{B_k(s)}{D(s)}} = \frac{N_0(s)}{D(s)} \frac{D(s)}{D(s) - B_k(s)} = \frac{N_0(s)}{D(s) - B_k(s)}. \quad (4.5)$$

The pole-zero cancellation is stable and does not affect the stability of the system. The state-space equivalent matrices can be computed from the transfer functions which directly define the digital filters. The matrices f_k , \underline{g} and h_k can all be defined from Equation (4.5) using Equation (3.9). Then matrices \bar{f}_k , \bar{g} and \bar{h}_k can be defined from Equations (4.3) and (4.4). At this point, the controller is well defined, but an observer needs to be added. The observer estimates the mirror height displacement position at each mass position since in the real system the true mirror height displacement is not known. The control signal \bar{u}_k is computed by combining a state estimator and state feedback as

$$\hat{\dot{x}}_k = \bar{f}_k \hat{x}_k + \bar{g} \bar{u}_k + K_k (y_k - \bar{h}_k \hat{x}_k) \quad (4.6)$$

and

$$\bar{u}_k = -L_k \hat{x}_k \quad (4.7)$$

with K_k such that $\det(sI - \bar{f}_k + K_k \bar{h}_k) \doteq \Gamma(s)$ and L_k such that $\det(sI - \bar{f}_k + \bar{g} L_k) \doteq \Delta(s)$. Both of the polynomials $\Gamma(s)$ and $\Delta(s)$ should be exactly the same for all \underline{k} in $\langle k_1, k_2 \rangle$. In order to accomplish this, the values for K_k and L_k must be solved for each \underline{k} by selecting the pole placement locations. These placements can be made to simulate the expected real-world system

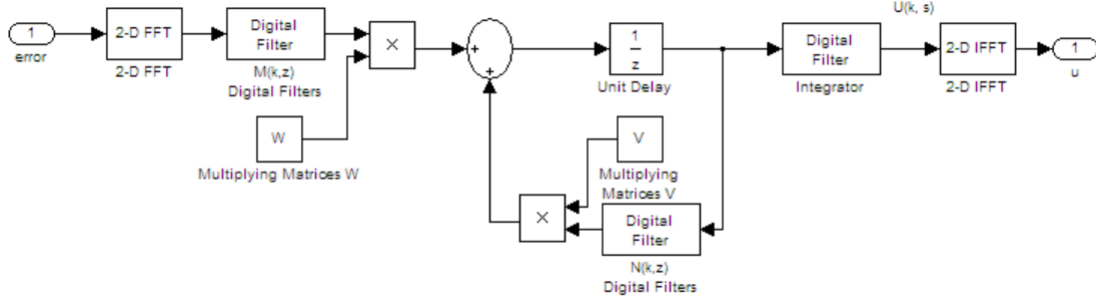


Figure 4.6: Controller model for deformable mirror.

with additional information from the SMT.

All of the system developments up to now lead to the creation of the system shown in Figure 4.6. The information needed to produce the combination feed-forward and feedback system that can control the DM is highlighted by this model. The observer is made to follow the rule of thumb that it converge to the actual state at four to ten times faster than the state changes in the pole selection. The number of poles placed in both the controller and observer equals the order of the system, which in this case is three. In the figure, the W and V matrices represent the polynomial coefficients of the transfer function numerator terms. There are three each, while the denominator is a fourth order polynomial that is the same for all k .

Combining Figures 4.5 and 4.6, we get the resulting complete simulation shown in Figure 4.7. The “phi” data input into the system shown in Figure 4.7 is from a phase generation program run externally before the simulation [14]. Ideally, in the real system, the wavefront reconstruction algorithm discussed in this thesis can be used to generate the estimated phase.

The simulation is configured with demonstration parameters that exhibit the dynamics expected from a DM. The resulting system is stable, and the time-varying wavefront is fed into the system controller which provides the actuator displacements to the mirror.

In Figure 4.8, the first 25 wavefront updates are shown, and the single pixel maximum error is plotted. The controller is adjusting the actuators to correct for the error as shown in the plot. The wavefront is sampled at a lower rate than the actuators can be moved, so the saw-tooth pattern shows that the error is approaching zero. When the new wavefront is provided, the maximum pixel error has a discontinuous jump, and the pattern keeps repeating. The percentage is calculated based on the absolute maximum phase value over the entire aperture. This information can be treated as a figure of merit to show that the controller is working across the large set of

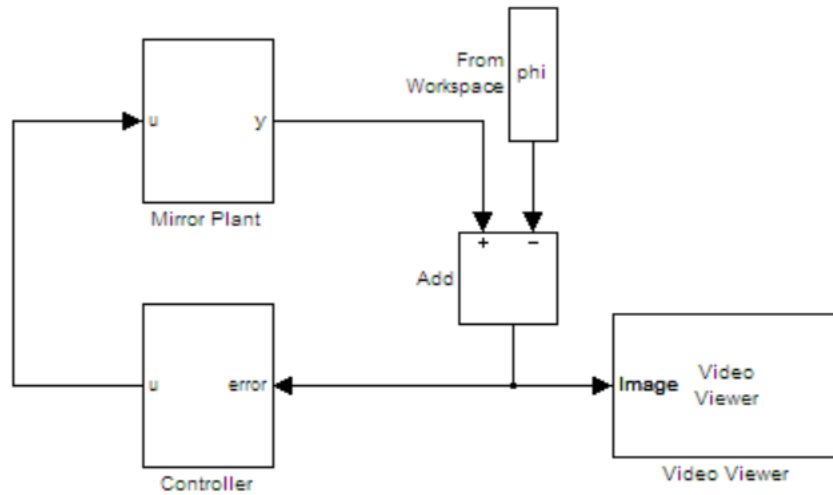


Figure 4.7: Simulation model for deformable mirror controller.

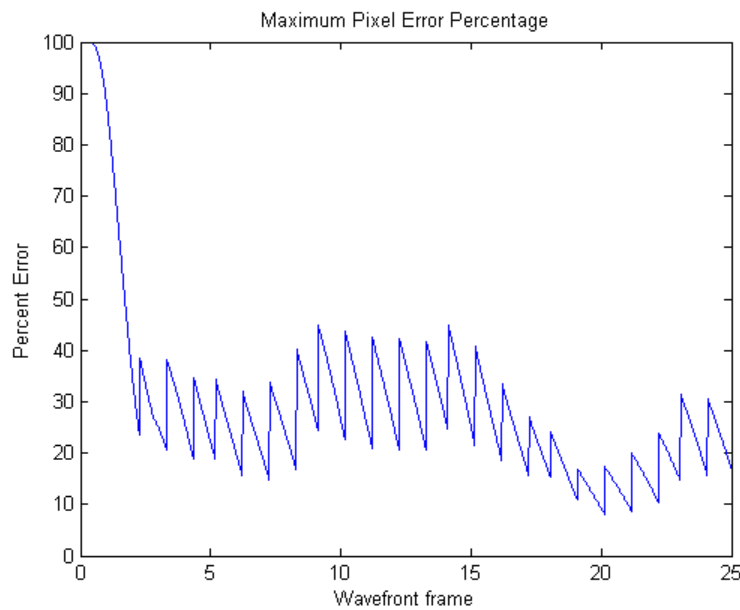


Figure 4.8: Maximum pixel error across the entire aperture for 25 wavefront updates.

data.

In Figure 4.9, the pixel error is broken up into the difference between the maximum and the minimum. This shows the error envelope. The negative error is much larger initially from the large phase changes in the blue regions shown in Figure 1.2. The error on both sides now has

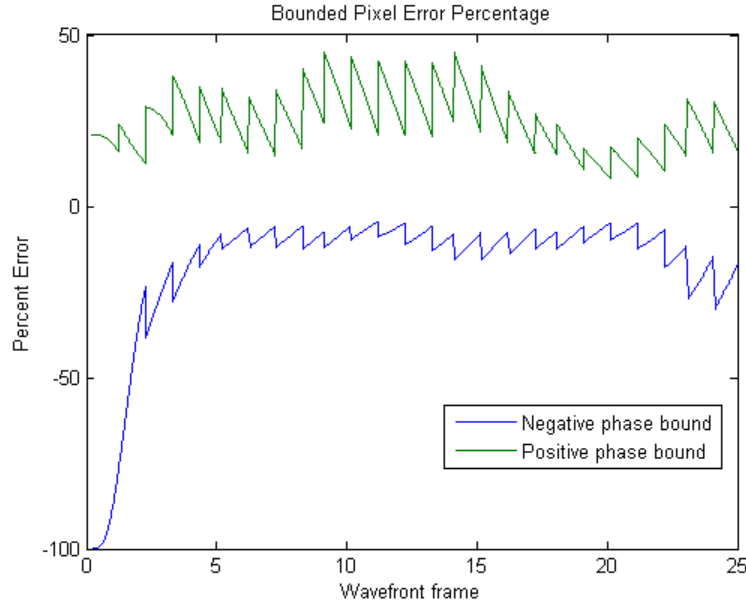


Figure 4.9: Maximum error extents on either side of the wavefront across the entire aperture for 25 wavefront updates.

the saw-tooth pattern. At the beginning of the plot, the lower side error makes large jumps from larger changes in the example wavefront. Despite the large changes, the system responds by a much sharper rate of error decrease. This is an interesting point. The graph results are not tracking the same individual pixels throughout the simulation but rather the specific pixels with maximum error extents.

In the spatial frequency domain, the majority of the wavefront energy is concentrated in the lower frequencies. This detail can be exploited further to minimize the required computing of all spatial frequencies to just the range of interest. If the system can tolerate the higher frequency content being uncontrolled, the computing requirements will decrease. In this manner, extremely large apertures can be processed without higher end computing resources. This can also be exploited on a space-based telescope, where the processors lag ground-based performance by nearly a decade.

THIS PAGE INTENTIONALLY LEFT BLANK

CHAPTER 5:

Conclusions

5.1 Summary

The main focus of this thesis is the application of the SMT to adaptive optics. In particular it addressed two important issues: wavefront reconstruction and feedback control. Wavefront reconstruction is developed based on the fundamental geometry of the data and the grid points where the wavefront phase is measured. Interpolating between these points, we compute the estimate of the wavefront. Techniques vary by computational complexity. While ground-based telescopes can have large computing power, space-based telescopes usually have other limiting factors of size, weight, power and electronics reliability to consider. While the industrial community is making larger apertures with increased actuator density, the algorithm's computational requirements should not scale faster than the computing capabilities of the platform in future systems.

The algorithms presented in this research are based on the DFT, which is computationally efficient and offers results comparable to other techniques such as the Zernike Polynomials. The wavefront can be reconstructed for non-noisy data or with a least-squares fit for noisy data.

Similarly, the control is based on a simple model of the DM as a mesh of masses connected by springs. The mirror has actuators along the normal of the rear surface that adjust the heights. The model is developed from basic theory and has not yet been adopted yet to the SMT. By assuming linearity and stationarity in the time and spatial domains, we can take full advantage of a decomposition in terms of spatial frequencies using the FFT2. In this way, the highly connected time-space domain model is reduced to a set of independent models in the frequency domain which can be easily controlled.

5.2 Additional Work

The wavefront reconstruction was accomplished for the single hexagon panel. Additional work could apply this to the six panels of the SMT. The interesting contribution of this would be to determine if the panels should be solved independently or as a larger more complicated aperture to see under which circumstances would the error would improve.

In addition, there are a number of tests to run on the actual telescope hardware. The effect of mechanical vignetting of the SMT in wavefront reconstruction is not known. The relationship of the outer edge of the hexagon aperture to the mirror's flat edges to the lenslet jagged hexagon edges, seen in Figure 3.1, is not known. The lenslets may all be contained inside of the mirror edge or extend past the edge. If there is overlap, it should be studied to determine the effect on wavefront reconstruction.

There has been previous work done by Puschel and Rotteler on the discrete Triangle Transform (DTT) [16]. This work could be applied to the truss lattice of the actuators. Additionally, Vince and Zheng have worked on taking the DFT of a hexagon lattice [17]. This work could be applied to the wavefront sensor lattice. Future work could compare which method allows for an efficient transformation between lattices and solve the varying relationships for the actuators to wavefront sensors in the lattices. There could be interesting consequences of the DTT approach given that it is similar to the discrete Cosine Transform (DCT-III) and is less sensitive to boundary conditions. A study of how the data processing requirements vary from this technique would be beneficial.

An interesting consequence of the hexagonal lenslets is that they are not symmetric about the two-slope axis. This could have interesting effects on the wavefront reconstruction and secondary camera, a topic worth exploring with additional research.

APPENDIX A:

Equation Set for Circular Aperture

The following equations represent a solution to the boundary conditions represented by Equations (2.8) for Figure 2.3:

$$\begin{aligned}
 u_2 - u_1 &= 0 & u_3 - s_x[3, 2] - u_2 &= 0 \\
 u_4 + s_y[5, 1] - s_x[4, 2] - u_3 &= 0 & u_5 + u_6 - s_x[5, 2] - s_y[5, 1] &= 0 \\
 u_7 - s_x[6, 2] - u_6 &= 0 & -u_8 - u_7 &= 0 \\
 u_8 + u_9 - s_x[7, 3] - s_y[7, 2] &= 0 & -u_{10} - u_9 &= 0 \\
 u_{10} - u_{11} - s_y[8, 3] &= 0 & u_{11} + u_{12} - s_x[8, 5] - s_y[8, 4] &= 0 \\
 s_x[8, 5] + u_{13} - u_{14} - s_y[8, 5] &= 0 & u_{14} - u_{15} - s_y[8, 6] &= 0 \\
 u_{15} - u_{16} &= 0 & s_x[7, 7] + u_{16} - u_{17} - s_y[7, 7] &= 0 \\
 u_{17} - u_{18} &= 0 & s_x[6, 8] + u_{18} - u_{19} &= 0 \\
 s_x[5, 8] + u_{19} - u_{20} - s_y[5, 8] &= 0 & s_x[4, 8] + s_y[5, 8] - u_{21} - u_{22} &= 0 \\
 s_x[3, 8] + u_{22} - u_{23} &= 0 & u_{24} + u_{23} &= 0 \\
 s_x[2, 7] + s_y[3, 7] - u_{24} - u_{25} &= 0 & u_{26} + u_{25} &= 0 \\
 u_{27} + s_y[2, 6] - u_{26} &= 0 & s_x[1, 5] + s_y[2, 5] - u_{27} - u_{28} &= 0 \\
 u_{30} + s_y[2, 4] - s_x[1, 5] - u_{29} &= 0 & u_{31} + s_y[2, 3] - u_{30} &= 0 \\
 -u_{32} - u_{31} &= 0 & u_1 + s_y[3, 2] - s_x[2, 3] - u_{32} &= 0
 \end{aligned}$$

These are shown for completeness to illustrate the example. These equations are created by writing the mesh equation for each square. The path followed along each square is top, right, bottom, left. If the slope arrow points in the opposite direction of this path, it is distinguished by negative sign.

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX B:

MATLAB code

B.1 ComputeSlopes.m

```
1 % This file implements an algorithm for solving the boundary slopes
2 % and then displays the results for a sample set of data.
3 % Useful output variables:
4 % unknowns - structure of the unknown slopes. fields are: axis, j, k
5 % equations - array of the unknowns for each equation (2 per eqn)
6 %   fields are: axis, j, k, sign
7 % knowns - array of known slopes needed for each equation (2 per row)
8 %   This is useful for dynamic, time varying data
9 %   Just iterate over this, sum the values per row for new c vector
10 %   fields are: axis, j, k, sign
11 % M, Minv - matrices that correspond to equations.
12
13 % code by Travis Axtell
14
15 close all;
16 clear all;
17 clc;
18
19 % load phi variable
20 load data;
21
22 % pad zeros on all sides to guarantee an edge
23 [size1, size2] = size(phi);
24 phi2 = zeros(size1+2,size2+2);
25 phi2(2:(size1+1),2:(size2+1)) = phi;
26 phi = phi2;
27 clear phi2;
28
29 % Find the aperture of data. This code depends on the actual data.
30 % For this example data, the edge is entirely surrounded by 0.
31 % This should be confirmed before applying new data to algorithm.
32 aperture = phi ~= 0;
33
```

```

34 % Find the slopes from the example phase data.
35 % Real world data would just be provided without knowing the phase.
36 sx = [phi(:, 2:size1+2), phi(:, 1)] - phi;
37 sy = [phi(2:size1+2, :); phi(1, :)] - phi;
38
39 % Empty variables to begin the code.
40 unknowns = []; % list of structures that identify the unknown slopes
41 knowns = []; % list of structures that identify the known slopes
42 sums = []; % vector of sums, c, to solve for u.
43 equations = []; % list of equations using the unknowns
44
45 % Placeholder for zeros as knowns.
46 kzero = struct('axis', 'zero', 'j', j, 'k', k, 'sign', 1);
47
48 % iterate over each square of values from the aperture
49 for j = 1:size1+1,
50     for k = 1:size1+1,
51         % binary value for the arrangement of the 4 values are as shown:
52         % 1 2
53         % 4 8
54         value = aperture(j,k) + 2*aperture(j,k+1) + ...
55                 4*aperture(j+1,k) + 8*aperture(j+1,k+1);
56
57         % This switch statement is not truly required but helps to organize
58         % the code.
59         switch value
60             case { 0, 15 }
61                 % the block is all internal or external and no
62                 % processing is required.
63                 continue;
64             case { 1, 2, 4, 8 }
65                 % single point on aperture.
66                 % Each if statement below identifies the two unknowns and
67                 % their associated mesh equation sign.
68                 sum = 0;
69                 % k1, k2 are placeholders since they are zeros.
70                 k1 = kzero;
71                 k2 = kzero;
72                 if value == 1,
73                     u1 = struct('axis', 'x', 'j', j, 'k', k, 'sign', 1);
74                     u2 = struct('axis', 'y', 'j', j, 'k', k, 'sign', -1);

```



```

75     elseif value == 2,
76         u1 = struct('axis', 'x', 'j', j, 'k', k, 'sign', 1);
77         u2 = struct('axis', 'y', 'j', j, 'k', k+1, 'sign', 1);
78     elseif value == 4,
79         u1 = struct('axis', 'y', 'j', j, 'k', k, 'sign', -1);
80         u2 = struct('axis', 'x', 'j', j+1, 'k', k, 'sign', -1);
81     elseif value == 8,
82         u1 = struct('axis', 'x', 'j', j+1, 'k', k, 'sign', -1);
83         u2 = struct('axis', 'y', 'j', j, 'k', k+1, 'sign', 1);
84     end
85     % checkadd confirms the unknown is new before adding it to
86     % the unknowns list, or if it is not new, redefines the u1
87     % to have the correct unknown number for generating the
88     % equations.
89     [unknowns, u1] = checkadd(unknowns, u1);
90     [unknowns, u2] = checkadd(unknowns, u2);
91
92     case { 3, 5, 10, 12 }
93         % two points on aperture.
94         sum = 0;
95         % Each if statement below identifies the two unknowns and
96         % their associated mesh equation sign.
97         if value == 3,
98             u1 = struct('axis', 'y', 'j', j, 'k', k, 'sign', -1);
99             u2 = struct('axis', 'y', 'j', j, 'k', k+1, 'sign', 1);
100            sum = sx(j,k);
101            k1 = struct('axis', 'x', 'j', j, 'k', k, 'sign', 1);
102            k2 = kzero;
103        elseif value == 5,
104            u1 = struct('axis', 'x', 'j', j, 'k', k, 'sign', 1);
105            u2 = struct('axis', 'x', 'j', j+1, 'k', k, 'sign', -1);
106            sum = -sy(j,k); % negative for mesh equation
107            k1 = struct('axis', 'y', 'j', j, 'k', k, 'sign', -1);
108            k2 = kzero;
109        elseif value == 10,
110            u1 = struct('axis', 'x', 'j', j, 'k', k, 'sign', 1);
111            u2 = struct('axis', 'x', 'j', j+1, 'k', k, 'sign', -1);
112            sum = sy(j,k+1);
113            k1 = struct('axis', 'y', 'j', j, 'k', k+1, 'sign', 1);
114            k2 = kzero;
115        elseif value == 12,

```

```

116         u1 = struct('axis', 'y', 'j', j, 'k', k, 'sign', -1);
117         u2 = struct('axis', 'y', 'j', j, 'k', k+1, 'sign', 1);
118         sum = -sx(j+1, k); % negative for mesh equation
119         k1 = struct('axis', 'x', 'j', j+1, 'k', k, 'sign', -1);
120         k2 = kzero;
121     end
122     % checkadd confirms the unknown is new before adding it to
123     % the unknowns list, or if it is not new, redefines the u1
124     % to have the correct unknown number for generating the
125     % equations.
126     [unknowns, u1] = checkadd(unknowns, u1);
127     [unknowns, u2] = checkadd(unknowns, u2);
128
129     case { 7, 11, 13, 14 }
130         % three points on aperture.
131         sum = 0;
132         % Each if statement below identifies the two unknowns and
133         % their associated mesh equation sign.
134         if value == 7,
135             u1 = struct('axis', 'x', 'j', j+1, 'k', k, 'sign', -1);
136             u2 = struct('axis', 'y', 'j', j, 'k', k+1, 'sign', 1);
137             sum = sx(j,k) - sy(j,k); % signs for mesh equation
138             k1 = struct('axis', 'x', 'j', j, 'k', k, 'sign', 1);
139             k2 = struct('axis', 'y', 'j', j, 'k', k, 'sign', -1);
140         elseif value == 11,
141             u1 = struct('axis', 'y', 'j', j, 'k', k, 'sign', -1);
142             u2 = struct('axis', 'x', 'j', j+1, 'k', k, 'sign', -1);
143             sum = sx(j,k) + sy(j,k+1); % signs for mesh equation
144             k1 = struct('axis', 'x', 'j', j, 'k', k, 'sign', 1);
145             k2 = struct('axis', 'y', 'j', j, 'k', k+1, 'sign', 1);
146         elseif value == 13,
147             u1 = struct('axis', 'x', 'j', j, 'k', k, 'sign', 1);
148             u2 = struct('axis', 'y', 'j', j, 'k', k+1, 'sign', 1);
149             sum = -sy(j,k) - sx(j+1,k); % signs for mesh equation
150             k1 = struct('axis', 'y', 'j', j, 'k', k, 'sign', -1);
151             k2 = struct('axis', 'x', 'j', j+1, 'k', k, 'sign', -1);
152         elseif value == 14,
153             u1 = struct('axis', 'x', 'j', j, 'k', k, 'sign', 1);
154             u2 = struct('axis', 'y', 'j', j, 'k', k, 'sign', -1);
155             sum = -sx(j+1, k) + sy(j,k+1); % signs for mesh equation
156             k1 = struct('axis', 'x', 'j', j+1, 'k', k, 'sign', -1);

```

```

157         k2 = struct('axis', 'y', 'j', j, 'k', k+1, 'sign', 1);
158     end
159     % checkadd confirms the unknown is new before adding it to
160     % the unknowns list, or if it is not new, redefines the u1
161     % to have the correct unknown number for generating the
162     % equations.
163     [unknowns, u1] = checkadd(unknowns, u1);
164     [unknowns, u2] = checkadd(unknowns, u2);
165
166     otherwise
167         % This code should never get executed since all cases are
168         % handled, but is included to verify that the code executes
169         % correctly.
170         % W: for warning
171         fprintf('W: (j, k) = (%d, %d); value = %d\n', j, k, value);
172     end
173
174     % The lists are updated to include the new entries
175     equations = [equations; u1, u2];
176     sums = [ sums; sum ];
177     knowns = [ knowns; k1, k2 ];
178 end
179 end
180
181 % Develop the M matrix from the equations
182 eqnlen = length(equations);
183 M = zeros(eqnlen,eqnlen);
184
185 for m = 1:eqnlen,
186     % Each row of M contains 1 equation, which includes 2 entries.
187     M(m, equations(m,1).number) = equations(m,1).sign;
188     M(m, equations(m,2).number) = equations(m,2).sign;
189 end
190
191 % If the Minv is needed to be computed, uncomment this
192 %Minv = pinv(M);
193 % In this case, the Minv was precomputed. This is how it would be done in
194 % actual implementation for faster results.
195 load Minv.mat;
196
197 % Solving for u is now a simple statement.

```

```

198 % For dyanmic data, rebuild sums using knowns before this next line.
199 u = Minv*sums;
200
201 % Now have a solution. Just need to apply it to the original data.
202 % sx = ix + bx => ix = sx - bx
203 % ix internal slope, sx computed slope, bx boundary slope
204 % ix is what is needed, sx is what was measured, bx is the error to remove
205 ix = sx;
206 iy = sy;
207
208 unklen = length(unknowns);
209 for m = 1:unklen,
210     % This if statement determines which set of data to apply the solved
211     % unknown to.
212     if unknowns(m).axis == 'x',
213         ix(unknowns(m).j, unknowns(m).k) = ...
214             sx(unknowns(m).j, unknowns(m).k) - u(m);
215     elseif unknowns(m).axis == 'y',
216         iy(unknowns(m).j, unknowns(m).k) = ...
217             sy(unknowns(m).j, unknowns(m).k) - u(m);
218     end
219 end
220
221 % The mean is removed from the data before its FFT2 is taken.
222 sXfft = fft2(ix-mean(mean(ix)));
223 sYfft = fft2(iy-mean(mean(iy)));
224
225 N = size1+2;
226 % denom2 is only used to make sure the denom is taking on a good value.
227 %denom2 = zeros(N,N);
228 PHI = zeros(N,N); % preallocated for speed
229 for j = 1:N,
230     for k = 1:N,
231         term1 = (exp(-1j * 2 * pi * (k-1)/N) - 1) * sXfft(j, k);
232         term2 = (exp(-1j * 2 * pi * (j-1)/N) - 1) * sYfft(j, k);
233         denom = 4 * (sin(pi * (j-1)/N)^2 + sin(pi * (k-1)/N)^2);
234         if denom < 0.000000005,
235             % this denominator is getting to be a small value, but we
236             % cannot divide by zero. So this checks to prevent that from
237             % occurring. Choosing this threshold is based on the N
238             % dimension. For larger N, the threshold must be smaller.

```

```

239         % This has major changes on the outcome of the reconstruction.
240         denom = 0.000000001;
241     end
242     %denom2(j, k) = denom;
243
244     % This PHI equation is from the Poyneer paper.
245     PHI(j, k) = (term1 + term2) / denom;
246 end
247 end
248
249 % The Poyneer paper specifically states the (1,1) entry needs to be 0, but
250 % in this filter implementation, the 0 is always there anyways.
251 %PHI(1,1)=0;
252
253 % the reconstructed wavefront. Should be completely real but it does have
254 % some small imaginary components from roundoff errors.
255 phinew = ifft2(PHI);
256
257 phimean = phi-mean(mean(phi));
258
259 % Error plot
260 figure; imagesc(abs(phimean-phinew));
261 title('Error between estimate and actual'); axis off;
262
263 % Original plot
264 figure;imagesc(real(phimean));
265 title('\Phi original data (mean subtracted)'); axis off;
266
267 % Reconstructed plot
268 figure; imagesc(real(phinew));
269 title('\Phi estimate data'); axis off;

```

B.2 CheckAdd.m

```

1 function [ list, entry ] = checkadd( inlist, inentry )
2 %[list, entry] = CHECKADD(inlist, inentry)
3 % Selectively adds to unknowns list
4 % If variable inentry is contained in the variable inlist,
5 % the output is an updated entry with the correct .number
6 % and the list is not changed.

```

```

7 %
8 % If the variable inentry is not in the variable inlist, it is added and a
9 % new number is assigned. The updated list and entry are the output.
10
11 len = length(inlist);
12 add = 1;
13 temp = -1;
14
15 for j = 1:len,
16     if (inlist(j).axis == inentry.axis) && ...
17         (inlist(j).j == inentry.j) && ...
18         (inlist(j).k == inentry.k)
19         % this entry already exists, so don't add it.
20         add = 0;
21         temp = j; % set the output to the existing entry
22     end
23 end
24
25 if add == 1,
26     % add the unknown number
27     inentry.number = len+1;
28     % set the list to include the new entry
29     % the .sign is removed as it is not important for the list purposes.
30     list = [inlist, rmfield(inentry, 'sign')];
31 else
32     % list is not updated to include new entry
33     list = inlist;
34     if temp > -1,
35         inentry.number = temp;
36     end
37 end
38
39 % set entry
40 entry = inentry;
41 end

```

APPENDIX C:

Additional Figures

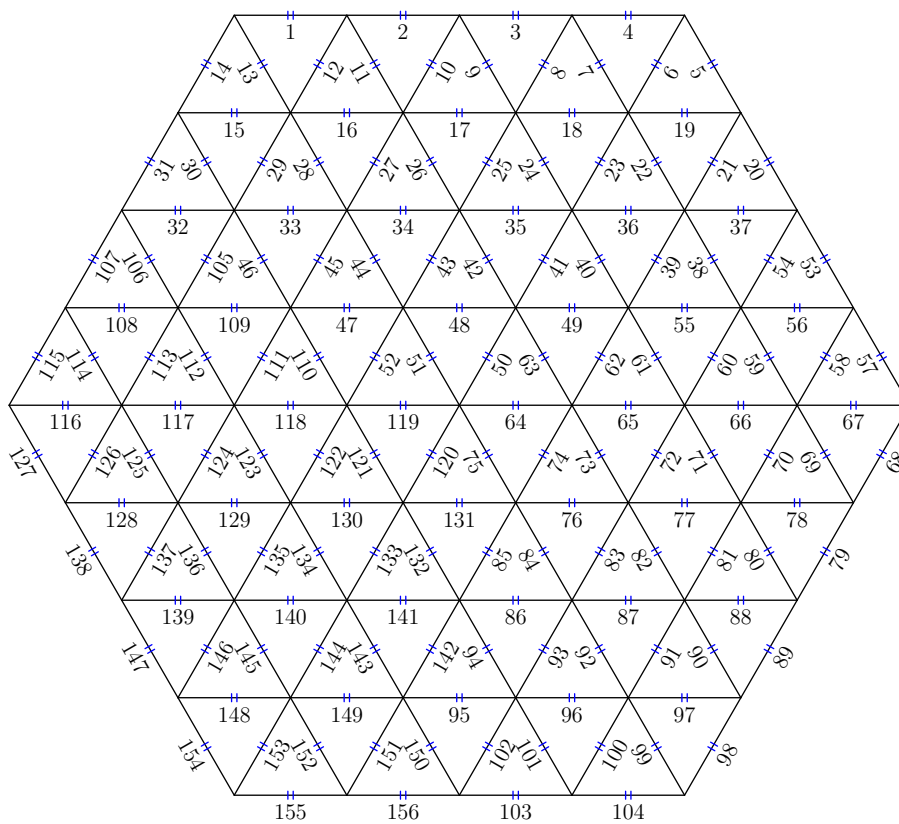


Figure C.1: SMT numbered actuator topology for a single segment.

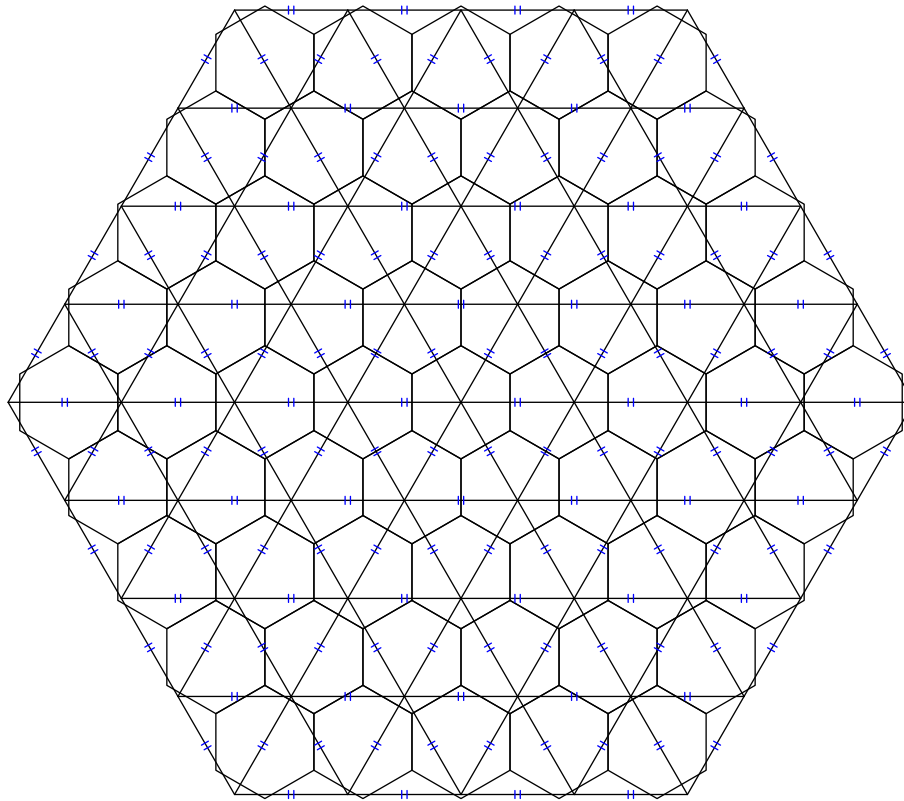


Figure C.2: SMT wavefront sensor and actuator topology for a single segment.

Index

active-optics system, 2, 18
actuator, 1, 2, 4, 5, 7, 8, 12, 15–19, 27, 31, 32
adaptive-optics system, 2, 18, 21, 31

controller, 2–4, 8, 15, 18–20, 25–27

deformable mirror, 2–4, 8, 15–18, 23, 27, 31
Discrete Fourier Transform, 3, 6, 7, 15, 31, 32
Discrete Spatial Fourier Transform, 23, 25

Fast Fourier Transform, 3, 7, 15, 21, 23, 25, 31

index of refraction, 5

James Webb Space Telescope, 12

National Reconnaissance Office, 1
National Security Space Strategy, 2
Naval Postgraduate School, 1, 21

optical path length, 5

phase points, 6–8, 10, 12, 13, 15, 21, 22
piston, 10

Segmented Mirror Telescope, 1–4, 6, 12, 15,
27, 31, 32, 44, 45
Shack-Hartmann sensor, 5, 6
state-space equation, 18, 19, 25, 26
state-space model, 19, 20

wavefront reconstruction, 3–5, 9, 21, 23, 27,
31, 32
wavefront sensor, 2, 5, 6, 12, 15, 32

Zernike polynomial, 4, 6, 31

REFERENCES

- [1] J. P. Gardner, J. C. Mather, M. Clampin, R. Doyon, M. A. Greenhouse, H. B. Hammel, J. B. Hutchings, P. Jakobsen, S. J. Lilly, K. S. Long, J. I. Lunine, M. J. McCaughrean, M. Mountain, H. walter Rix, E. P. Smith, G. Sonneborn, M. Stiavelli, H. S. Stockman, R. A. Windhorst, and G. S. Wright, “The James Webb space telescope,” *Space Science Reviews*, vol. 123, no. 4, pp. 485 – 606, 2006.
- [2] B. Honneger, “NPS new home for giant segmented-mirror space telescope.” Internet, January 2010. <http://www.nps.edu/About/News/NPS-New-Home-for-Giant-Segmented-Mirror-Space-Telescope-.html>.
- [3] R. Tyson, *Introduction to Adaptive Optics*. Bellingham: SPIE Press, 2000.
- [4] “National Security Space Strategy unclassified summary,” January 2011.
- [5] K. Freischlad, “Wavefront reconstruction from noisy slope or difference data using the discrete Fourier transform.,” vol. 551, (Arlington, VA, USA), pp. 74 – 80, 1985.
- [6] L. A. Poyneer, D. T. Gavel, and J. M. Brase, “Fast wave-front reconstruction in large adaptive optics systems with use of the Fourier transform,” *Journal of the Optical Society of America A: Optics and Image Science, and Vision*, vol. 19, no. 10, pp. 2100 – 2111, 2002.
- [7] R. H. Hudgin, “Wave-front reconstruction for compensated imaging,” *J. Opt. Soc. Am.*, vol. 67, pp. 375–378, Mar 1977.
- [8] D. L. Fried, “Least-square fitting a wave-front distortion estimate to an array of phase-difference measurements,” *J. Opt. Soc. Am.*, vol. 67, pp. 370–375, Mar 1977.
- [9] F. Roddier, *Adaptive Optics in Astronomy*. Cambridge: Cambridge University Press, 1999.
- [10] F. Pedrotti, *Introduction to Optics*. Englewood Cliffs N.J.: Prentice Hall, 2nd ed. ed., 1993.
- [11] J. M. Geary, *Introduction to Wavefront Sensors*. Bellingham: SPIE Optical Engineering Press, 1995.

- [12] B. C. Platt and R. Shack, “History and principles of Shack-Hartmann wavefront sensing,” *Journal of refractive surgery (Thorofare, N.J. : 1995)*, vol. 17, no. 5, 2001.
- [13] R. J. Noll, “Zernike polynomials and atmospheric turbulence.,” *Journal of the Optical Society of America*, vol. 66, no. 3, pp. 207 – 211, 1976.
- [14] M. Corley, *Maritime Adaptive Optics Beam Control*. PhD thesis, Naval Postgraduate School, September 2010.
- [15] C. C. Wilcox, *Optical phase aberration generation using a liquid crystal spatial light modulator*. PhD thesis, University of New Mexico, 2009.
- [16] M. Puschel and M. Rotteler, “The discrete triangle transform,” in *none*, vol. 3, (Montreal, Que, Canada), pp. III45 – III48, 2004.
- [17] A. Vince and X. Zheng, “Computing the discrete Fourier transform on a hexagonal lattice,” *Journal of Mathematical Imaging and Vision*, vol. 28, no. 2, pp. 125 – 133, 2007.

Initial Distribution List

1. Defense Technical Information Center
Ft. Belvoir, Virginia
2. Dudley Knox Library
Naval Postgraduate School
Monterey, California
3. Norris C. Mitchell
Space and Naval Warfare Systems Center Atlantic
Charleston, South Carolina
4. Al Emondi
Space and Naval Warfare Systems Center Atlantic
Charleston, South Carolina
5. Clark Robertson
Naval Postgraduate School
Monterey, California
6. Roberto Cristi
Naval Postgraduate School
Monterey, California
7. Jae-Jun Kim
Naval Postgraduate School
Monterey, California
8. Travis Axtell
Naval Postgraduate School
Monterey, California